

TP 3 - Programmation mobile avancée

SANCHEZ Martin

Exercice 1 - Application de Quizz avec Firebase

L'objectif dans cet exercice était de reprendre l'application de quizz en stockant les questions dans une base de donnée firebase.

Pour cela j'ai repris l'application de quizz du tp précédent.



FIGURE 1 – Interface de l'application de quizz

Setup de Firebase

Pour ce tp j'ai donc utilisé une base de donnée hébergée sur Firebase, pour cela j'ai créé un nouveau projet que j'ai relié à mon application via la CLI de Firebase. Je n'ai pas fait comme c'était expliqué dans le cours, car j'utilise un navigateur pour tester mon application, l'exemple du cours prévoyait le lien avec une application Android.

Par la suite j'ai créé une base de donnée Firestore, dans laquelle j'ai ajouté une collection correspondant aux questions. Cette collection contiendra un ensemble de documents, chaque document représentera une question.

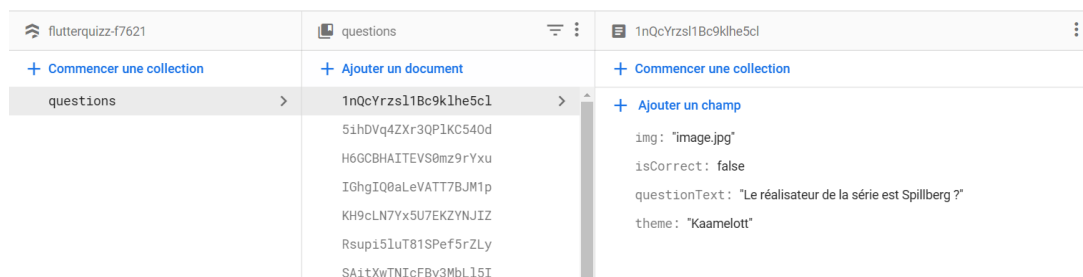


FIGURE 2 – Collection "question" dans le firestore.

Les interfaces de l'application

Même si l'application fonctionne de la même façon que dans les TP1 et 2, certains éléments ont été ajoutés, comme une page d'accueil qui permet de sélectionner les thèmes voulu pour le quizz, de se rendre sur le formulaire d'ajout de question, et de lancer le quizz.

Page d'accueil



FIGURE 3 – Page d'accueil de l'application.

Sur cette page on peut voir la liste des différents thèmes présents dans la base de donnée, cette liste est scrollable horizontalement. Cliquez sur les boutons pour sélectionner des thèmes, le bouton deviendra vert si le thème est sélectionné. Si aucun thème n'est choisis, tout les thèmes seront traité dans le quizz.

Le bouton "Jouer !" vous redirigera sur le quizz, et le boutons "Ajouter une question" sur le formulaire d'ajout.

Formulaire d'ajout de question

← Submitting question DEBUG

Back

Ajoutez votre question :

Entrez le texte de la question

Vrai Faux

Entrez le theme de la question

Entrez l'url de l'image (optionnel)

Ajouter !

FIGURE 4 – Formulaire d'ajout de l'application.

Ce formulaire contient des champs pour décrire une question. Le bouton ajouter enverra la question sur la base de données si elle valide les conditions suivantes :

- Le texte de la question n'est pas vide et contient un " ?".
- La réponse de la question est précisée.
- Le thème de la question n'est pas vide.

Interface du quizz

La dernière page est celle du quizz, la seule modification dans cette interface est que nous montrons le thème de la question haut dessus de l'image.

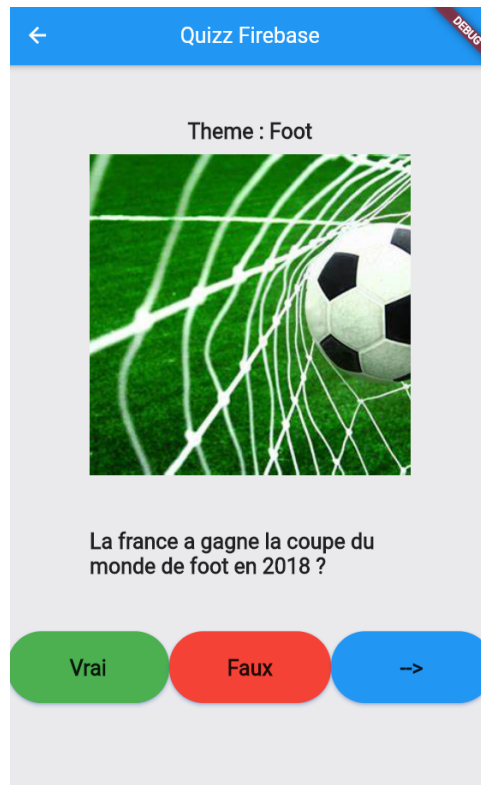


FIGURE 5 – Interface du quizz de l'application.

Utilisation de Firebase dans l'application

Ensuite pour accéder à la base de donnée dans l'application, j'ai du ajouter ou modifier plusieurs classes.

- Le main : Pour initialiser l'accès à la base de données.

```
13  ► void main() async {  
14      WidgetsFlutterBinding.ensureInitialized();  
15      await Firebase.initializeApp(  
16          options: DefaultFirebaseOptions.currentPlatform,  
17      );  
18      runApp(const MyApp());  
19  }
```

FIGURE 6 – Modifications de la classe main.

- Ajout d'une classe Database pour gérer les accès à la base de donnée ; en effet j'ai décidé d'encapsuler tout les accès a la base dans une seule classe.
- Le QuestionRepository ; qui à du être adapté pour aller chercher les données dans la classe Database.

La classe Database

Pour détailler un peut plus cette classe, elle s'occupe des accès à la base de donné. Elle implémente plusieurs méthodes statiques qui permettent d'ajouter et de récupérer les questions et les thèmes. Ces méthodes nous renvoient des Future

```
6  class Database {  
7  
8      static FirebaseFirestore getFirestore() {return FirebaseFirestore.instance; }  
9      static CollectionReference getQuestionsCollection() {      return getFirestore  
10  
11  |  static Future<void> addQuestion(Question q) {...}|  
16  
17  |  static Future<List<Question>> getQuestions(List<String> themes) async {...}|  
40  
41  |  static Future<List<String>> getThemes() async {...}|
```

FIGURE 7 – La classe Database.

Exercice 2 - Organisation du projet

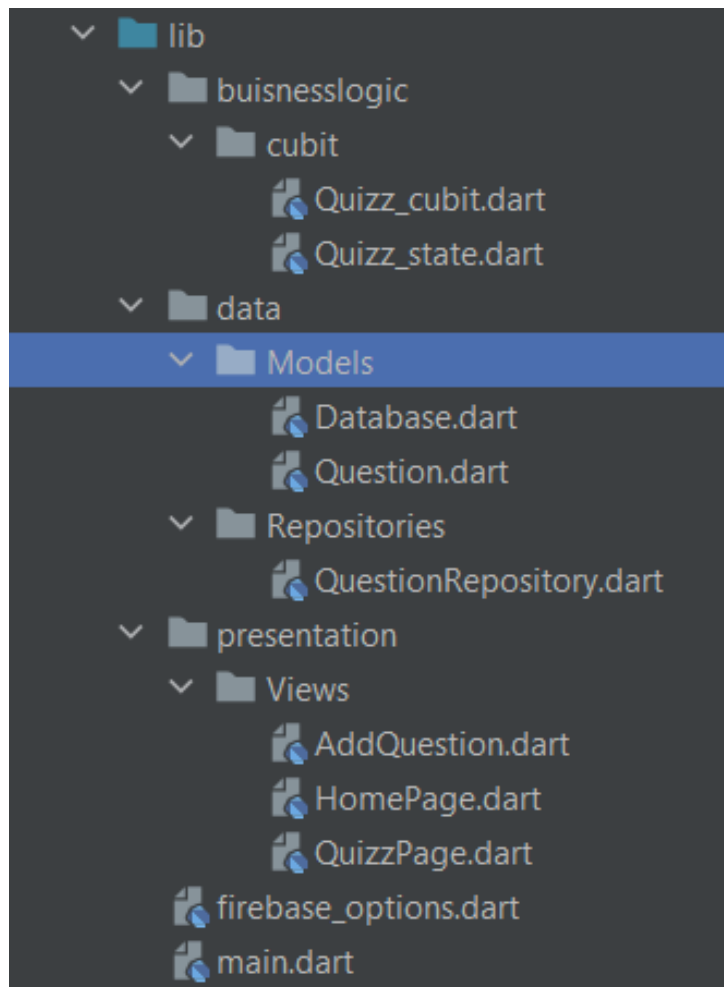


FIGURE 8 – Organisation des packages du projet.

Le fichier "firebase_option.dart" reste au même niveau que le main car il à été généré par la CLI de Firebase.